Optimisation algorithms for improvement of a multihead weighing process

Alexander Pulido-Rojano*
Department of Industrial Engineering,
Universidad Simón Bolívar,
Carrera 59 No. 59-92. A.A. 50595, Barranquilla, Colombia
E-mail: apulido3@unisimonbolivar.edu.co
*Corresponding author

J. Carlos García-Díaz
Centre for Quality and Change Management,
Universitat Politècnica de València,
Camino de Vera, s/n. 46022, Valencia, Spain
E-mail: juagardi@eio.upv.es

Biographical notes:

Alexander Pulido-Rojano received his B.S. Eng in Industrial Engineering in 2006 from the Universidad de la Costa in Barranquilla (Colombia), his M.S. degree in Industrial Engineering in 2012 from the Universidad del Norte (Colombia), and his PhD in Statistics and Optimization in 2017 from the Universitat Politècnica de València, Spain. He has worked on various research projects and he is currently a full Professor in the Industrial Engineering Program at the Universidad Simón Bolivar from Barranquilla, Colombia.

J. Carlos García-Díaz received his BS in 1991, MS in 1996 and his PhD in Applied Statistics in 2003 from the Polytechnic University of Valencia, Spain. Currently, he is a senior quality consultant and is an Associate Professor of statistical quality control at the Polytechnic University of Valencia.

Abstract

Mathematical optimisation is widely used to find the optimal value for an objective function, subject to constraints that try to simulate reality, and is fundamental to improving industrial processes. In this paper, we compare different optimization approaches to solve the packaging problem in multihead weighing machines. In this problem, each package is made up from the loads in a subset of the multihead weigher's hoppers. The total weight of the packed product must be as close to a specified target weight as possible. We designed and evaluated a set of algorithms for this problem, considering both single-objective and bi-objective optimization criteria. A new criterion for creating the packages is considered, and a different way of filling of the hoppers is studied with the aim of reducing process variability. Numerical experiments considering both a set of real data and the most important process performance parameters show the usefulness of our study.

Keywords: optimisation; mathematical modeling; exhaustive search; reduction of variability; process improvement; packaging; multihead weighing process.

1. Introduction

Optimisation is a fundamental discipline in fields such as information technology, artificial intelligence, and operations research. Optimisation is the process of trying to find the best possible solution to a problem, usually in a limited amount of time (Cook et al., 1998; Nemhauser and Wolsey, 1988). In an optimisation problem, there are many possible solutions and some clear way of comparing them to find the best one. In fact, such problems can be defined by the presence of a set of different candidate solutions that can be compared (Duarte et al., 2007). Depending on their algorithmic complexity, these problems can be categorized as either P, NP, NP-complete, or NP-hard (Bierlaire, 2015; Blum et al., 2008; Erdogdu, 2009; Marler, 2009).

For an important subset of optimisation problems, no exact algorithm is available that can find the optimal solution in a reasonable time. However, an alternative approach to solve these problems, is to design approximate algorithms that can find high-quality (though not necessarily optimal) solutions in a given time. Each problem is represented as a mathematical model comprising of an objective function and a set of constraints that somehow encode the optimisation problem (Bierlaire, 2015).

In this paper, we design and evaluate a set of approximate algorithms to solve the packaging problem for multihead weighing machines (i.e., the multihead weighing process). The algorithms are executed based on data obtained from real studies, following a proposed strategy for the filling of hoppers (Pulido-Rojano and García-Díaz, 2016). Also, our proposal encompasses both single-objective and bi-objective optimisation approaches.

From the point of view of improving the processes, to authors as Montgomery (2009), the improvement of the quality is the reduction of the variability in the processes and products. This definition implies that if the variability of the important characteristics of a product, process or service decreases, the quality of the product, process or service increases (Tejaskumar and Darshak, 2016; Sukrut and Mohammed, 2017). Therefore, increasing the competitiveness of a company is closely related to a continuous improvement of the quality in all its processes (Sasadhar and Indrajit, 2018; Selvam et al., 2018). In this sense, an important goal in this study is to analyse the performance of the proposed algorithms to improve the variability in the multihead weighing process.

This paper is structured as follows. Section 2 presents the packaging problem for multihead weighing machines. In Section 3, the proposed optimisation approach is described. The results of the numerical experiments are then presented and analyzed in Section 4, followed by our conclusions in Section 5.

2. Packaging problem

2.1 Multihead weighing machines

Multihead weighing machines are also known as combination weighing systems. They can be installed as part of an integrated weighing and packaging production line or interfaced with existing packaging equipment. They are ideally suited to precise and fast weighing of products, ranging in size from granulates to large and bulky products. Moreover, the machines can be used for almost all food products (dry, fresh, or deep frozen, such as snacks, crisps, sweets, fresh salad and vegetables), sea food, dog food, technical and pharmaceutical products. Multihead weighing machines use several different weighing techniques to obtain a total package weight (W) that is closer to the desired target weight (T) than can be achieved with conventional weighing techniques.

These weighing systems comprise three elements: a (linear) vibratory feeder system to automatically feed the product to the weighing stations; a system to collect the products and feed them to the feed hoppers; and a set of weighing hoppers that statically weigh the products, calculate all possible weight combinations, and dispense the best combination (closest match to the target weight) to a packaging machine. A detailed description of the arrangement of feeders and hoppers in a typical multihead weighing system can be found in (Pulido-Rojano et al., 2015), and Figure 1 shows the basic components of such a system.

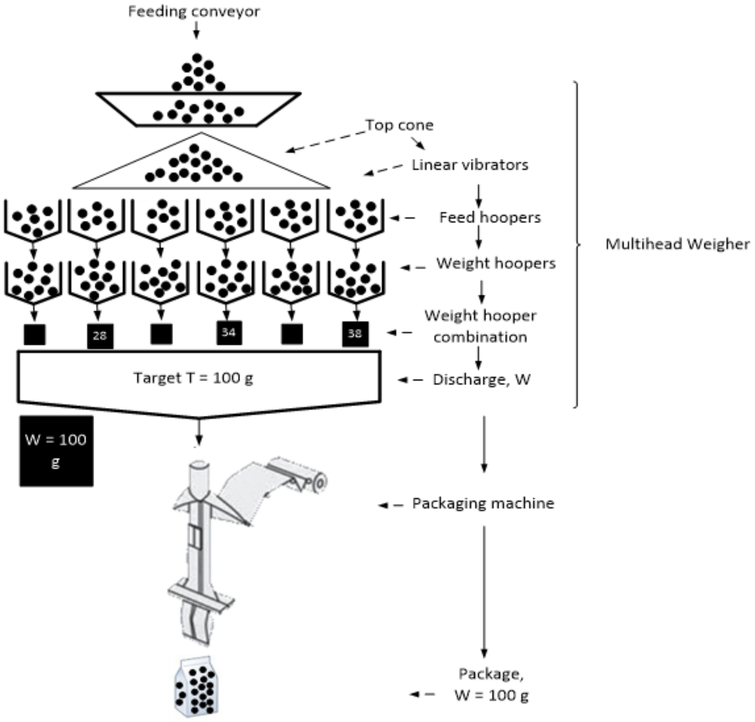


Figure 1. Basic components of a multihead weighing system. Source: García-Díaz and Pulido-Rojano (2017).

2.2. Multihead weighing process

In the multihead packaging process, a subset H' is chosen from the set H of products in the n current weighing hoppers to make up package. A quantity x_i of food is placed in each weighing hopper i $(i = 1,2, \ldots, n)$, and the weight signals from each hopper are transmitted to the built-in computer in the system's control unit. The computer then calculates all possible weights combinations, and the products in the subset that exceeds the desired weight T by the minimum possible amount are ejected from their corresponding hoppers. The resulting empty hoppers are then supplied with new quantities of food. The computer repeats this process until it has produced the number required of packages (Q). The number of different possible hopper subsets H' depends on the number k of hoppers that are selected for each packaging operation. This is equivalent to the NP-complete subset-sum combinatorial problem (Garey and Johnson, 1979) when k is neither predetermined nor constant.

The weights x_i $(i = 1,2, \ldots, n)$ in the hoppers follow a normal probability distribution. Based on a study that analysed real data, several authors (Beretta and Semeraro, 2012; Beretta et al., 2016; del Castillo et al., 2017) have noted that the weights x_i were normally distributed $x_i \sim \mathrm{N}(\mu, \sigma)$, where μ is the average weight of product supplied to the hoppers and σ is its standard deviation. When using a vibratory feeder, these quantities are correlated, in a way that depends on the form and weight of the product concerned. Some authors (Beretta and Semeraro, 2012; Beretta et al., 2016; del Castillo et al., 2017) have investigated these correlations and found that σ depends linearly on the mean weight μ, according to $\sigma = \gamma\mu$, where γ $(0 < \gamma < 1)$ is the proportionality coefficient for μ and σ, and depends on the product being packaged.

Note that if all the hoppers are filled independently according to the same distribution $N(\mu, \sigma = \gamma\mu)$, and the k hoppers used for each packaging operation are randomly selected (to make up a total weight around T), then the packages weights will also follow a normal distribution $N(k\mu, \sqrt{k}\sigma)$, where the mean package weight $k\mu$ is expected to equal the target T.

An additional point to consider is that a given quantity of product can remain in its corresponding hopper for a long time before being chosen for packaging. This can be a problem when handling products that can deteriorate quickly, such as, frozen goods. One possible way to tackle this problem is to monitor and control the time products spend in each hopper, which can be done by assigning a priority coefficient P_i to each hopper (Karuno et al., 2007). The priority P_i measures the time the current load has spent in hopper i and can be calculated as follows. Let ℓ denote the current iteration number of the packaging process, and let ℓ_i denote the iteration at which the current load was sent to the i-th hopper (i.e., the last time it was empty). Then, $P_i = \ell - \ell_i + 1$ represents the time (number of packaging operations) the load has spent in hopper i. Note that $1 \leq \ell \leq$ Q. In this context, we now require the packaging process to meet two objectives: make W as close to T as possible and minimize the total time the food in each package has spent in the packaging system.

This paper proposes to solve this problem by implementing approximate algorithms for cases where k is constant and predetermined. This means the average weight of product μ supplied to the n hoppers must be $\mu = T / k$. In particular, we consider two problems: minimizing the absolute difference between W and T and minimizing this value while also considering the residence time P_i. These problems are handled using a single-objective and bi-objective approaches, respectively, and evaluated considering a set of real data. Multihead weighing system manufacturers can then select and configure the approach that best fits their needs. Our goal is to demonstrate the usefulness of this approach for minimizing excess weight in the packages produced. The packaging problem will be formulated in terms of an allocation model, using binary variables to select the hoppers to use for each package (see Section 3.4).

The approach of minimizing of the absolute difference between W and T, and the idea of evaluating a fixed number of combined k-hoppers have already been studied by Pulido-Rojano and García-Díaz (2016) and García-Díaz et al. (2017), however, the authors assumed that the variability of the weights in the hoppers does not depend on the coefficient of proportionality γ, which would not be in line with industry practice. In addition, the authors presented this problem assuming the σ values to each hopper.

2.3. Related work

Several researchers have studied the possibility of improving multihead weighing and packaging processes through mathematical optimisation or approximate methods. For example, a percentage variability reduction index has been proposed (Barreiro et al., 1998; Salicrú et al., 1996) to reduce and control production process variability. The optimal scheme for determining the operation time of line feeders in automatic combination weighers has also been investigated (Keraita and Kim, 2006). A weighing algorithm for multihead weighers has been proposed (Keraita and Kim, 2007) that is based on bitwise operations. An additional objective, known as “priority” has also been introduced (Karuno et al., 2007). Here the problem was formulated as a bi-objective optimisation problem, and a dynamic programming algorithm was proposed for its solution. This algorithm aimed to minimize the maximum time items spent in the system heuristically, while also keeping the total weight of each package as close to the target weight as possible. Some authors (Imahori et al., 2011, 2012; Karuno et al., 2013; Karuno and Tateishi, 2014; Karuno and Saito, 2017) have studied the possibility of improving this bi-objective optimisation model. Other authors, such as (Imahori et al., 2012; Karuno et al., 2010), have investigated different types of packaging operations, developing several algorithms for double-layered and duplex packaging systems. Several optimisation algorithms have been proposed (Beretta et al., 2016) for determining the optimal flow rates for a set of radial feeders, with the objective of minimizing the expected production cost per “conforming” package over a fixed time period. The way the hoppers are filled has also been studied (Pulido-Rojano and García-Díaz, 2016) with the aim of reducing variability in the packaging process. An heuristic optimisation model has been developed (del Castillo et al., 2017), on the basis of a detailed characterization of what constitutes a near optimal solution to the multihead weigher setup problem. The idea was to find the set points for the hoppers that minimized the mean squared error of the package weight. The optimal operational conditions for the packaging process have also been obtained (García-Díaz et al., 2017) using a bi-objective algorithm. Finally, a modified control chart has been proposed (García-Díaz and Pulido-Rojano, 2017) for monitoring and controlling the multihead weighing process.

3. Optimisation approach

This section discusses the objectives considered for optimising the packaging process, the proposed method for determining the way the products are supplied to the weighing hoppers, the proposed algorithms for the packaging process itself, and the designed mathematical optimisation models.

3.1. Packaging process objectives

We use two optimisation approaches to address the packaging problem. The first, a single-objective approach, aims to minimize the absolute difference between the real package weight $W = \sum_{i \in H'} x_i$ and its target weight T. This can be expressed as: $z_1 = min|\ T - \sum_{i \in H'} x_i\ |$, where z_1 is the first objective.

In order to make this approach more realistic, we also include the following constraint proposed by Pulido-Rojano and García-Díaz (2016): $|\ T - \sum_{i \in H'} x_i\ | \leq \mathrm{Z}_{\alpha/2}\sqrt{k}\sigma,$ where $\mathrm{Z}_{\alpha/2}$ represents the critical value of the standard normal probability distribution $N(0,1)$ for a significance of α. This constraint (known as the *confidence level constraint*) avoids k-hopper subsets that would produce a package too far from the target T. In our case, the W value selected for each package must be within a confidence level of 99.73% of T, i.e., $Z\alpha/_2$= 3.0.

In the second (bi-objective) approach, we aim to minimize the difference between W and T as before ($z_1 = min|\ T - \sum_{i \in H'} x_i\ |$), while also maximizing the residence time P_i, as follows: $z_2 = max \sum_{i \in H'} P_i$. The goal of the second objective z_2 is to encourage the selection of hoppers that have not been emptied for a long time (i.e., with long residence times). To control how long the loads stay in the hoppers, we use P_{max} as the maximum number of packaging operations (i.e., the maximum allowed priority) for which any load is allowed to remain in its hopper (García-Díaz et al., 2017). For instance, if P_{max} = 100, the maximum time a load could remain in a multihead weigher with a capacity of fifty packages per minute can be calculated as follows: 50 packages / 60 s is equivalent to 1.2 s/package; thus, 1.2 s/package × 100 packages = 120 s. Based on this, a k-hopper subset is said to be valid if it does not involve any hoppers whose priorities are greater than the maximum allowed priority P_{max} and the total weight is in the range $T \pm Z\alpha/_2\sqrt{k}\sigma$.

For the bi-objective approach, we propose to use a single weighted performance function that combines information about the two objectives being considered, and dynamically adjust the relative weight or importance of each objective at each iteration of the packaging process, as suggested by García-Díaz et al. (2017). So, for each package we look for the k-hopper subset that minimizes the distance to the so-called *utopia* or *ideal point* $(\mathrm{z}_1^{\min}, z_2^{\max})$ in criterion space, where $z_1^{\min}$ is the minimum possible (absolute) difference between T and W, and $z_2^{\max}$ is the maximum possible aggregate (total) priority.

Essentially, $z_1^{\min}$ and $z_2^{\max}$ are the optimal values for the two objectives being considered for the current hopper's contents optimised separately. Prior to calculating the Euclidean distance (D) from a given solution to the ideal point, both values are normalized and then assigned relative weights of $(1 - \theta)$ and θ, respectively, so that the final form of the function to be minimized is:

$$D = \sqrt{(1-\theta)\left(\frac{z_1 - z_1^{\min}}{z_1^{\max} - z_1^{\min}}\right)^2 + \theta\left(\frac{z_2 - z_2^{\max}}{z_2^{\max} - z_2^{\min}}\right)^2}\ , \tag{1}$$

Here $z_1^{\max}$ and $z_2^{\min}$ are the maximum difference from the target weight and the minimum total priority, respectively, for the current set of valid k-hopper subsets.

The parameter θ is updated at each packaging iteration. The idea is that selecting a k-hopper subset with a high aggregate priority becomes more important as the current maximum hopper priority approaches the maximum allowed priority P_{max}. The θ value is defined as:

$$\theta = \frac{1}{P_{max} - \max_{i \in H} P_i + 1}. \quad (2)$$

So, during the first iterations, the value of θ will remain relatively small and, therefore, the objective of minimizing the difference to target packet weight will be assigned a higher importance. As packages production progresses and the maximum hopper priority approaches P_{max}, θ will increase and so will the importance of the priority objective increase.

The combination of hoppers that minimizes the distance to the ideal point is known to be an efficient or nondominated solution (Marler, 2009), which means that there is no other valid k-hopper subset that is at least as good with respect to (at least) one of the objectives (weight or priority) and strictly better with respect to the other objective (Ehrgott, 2005).

3.2. How to fill the hoppers

In this paper, we consider the general case where each hopper i is filled with a different average quantity of food μ_i (instead of a common value μ). The degree of variability among the average hopper weights $\mu_1 \ldots \mu_n$ is believed to be related to the final package variability. Here, we evaluate the case where groups of hoppers share the same μ_i value, as this has been shown to be an efficient strategy for reducing package variability (Barreiro et al., 1998; Keraita and Kim, 2007; Pulido-Rojano and García-Díaz, 2016).

To set the average amounts of product supplied to the weighing hoppers, we use shifts in the mean amount, given by the parameter δ. The parameter ensures that different average amounts are supplied to different hoppers. The purpose of introducing these deliberate shifts is to study the effect of changing the way the hoppers are filled.

As a strategy for setting the average amounts of product supplied to the hoppers, we propose to divide the n weighing hoppers into three subgroups (n_1, n_2, and n_3 with $n = \sum_{j=1}^{3} n_j$) and supply different average amounts of product to each subgroup (μ_1, μ_2 and μ_3, respectively) (Pulido-Rojano and García-Díaz, 2016). The average amounts supplied to each subgroup will depend on δ, as follows: $\mu_1 = \mu - \delta\sigma$, $\mu_2 = \mu$ and $\mu_3 = \mu + \delta\sigma$, respectively. Thus, shifts in the product supply will occur when $\delta > 0$, while if $\delta = 0$ then all the hoppers will be filled at the same rate, namely $\mu_j = \mu$, i.e., $\mu_1 = \mu_2 = \mu_3 = T/k$. In our case, the values of μ_1, μ_2 and μ_3 will also depend on $\mu = T/k$ and $\sigma = \gamma\mu$. Once the $\mu_j$values have been set, we can calculate the σ_j values as follows: $\sigma_j = \gamma\mu_j$. So far, no research has tested this filling strategy in a bi-objective approach. Authors as García-Díaz et al. (2017) tested a filling strategy for five subgroups.

The proportionality coefficient γ is used to calculate the standard deviations σ_j of the weights supplied to each hopper and is considered to be an input to the packaging process. In our numerical experiments, we will use the values given in Beretta and Semeraro (2012) for two specific products: "ravioli" (a type of dried pasta) with $\gamma = 0.331$ and "fusilli" (a type of fresh pasta) with $\gamma = 0.123$.

As an example of calculating the μ_j and σ_j values, suppose $T = 250$ g, $k = 5$, $\sigma = 16.55$ g (for $\gamma = 0.331$, i.e., the ravioli) and $\delta = 2.0$. In addition, suppose the total number of hoppers $n = 16$, with $n_1 = 5$, $n_2 = 6$ and $n_3 = 5$. Then, the μ_j values would be as follows: $\mu_1 = 250/5 - 2.0(16.55) = 16.90$ g, $\mu_2 = 250/5 = 50$ g, and $\mu_3 = 250/5 + 2.0(16.55) = 83.10$ g. In this case, the σ_j values would be $\sigma_1 = \gamma\mu_1 = 0.331 \cdot 16.90 = 5.59$ g, $\sigma_2 = \gamma\mu_2 = 0.331 \cdot 50 = 16.55$ g and $\sigma_3 = \gamma\mu_3 = 0.331 \cdot 83.10 = 27.51$ g.

3.3. Packaging algorithms

In this section, we introduce the proposed package production algorithms. These procedures, both single-objective and bi-objective, are performed for each package in order to find the k-hopper subset H' for which the total weight W is as close to the target weight T as possible (either above or below). As previously discussed, manufacturers could adapt this packaging algorithm for implementation in the control unit software of multihead weighers.

3.3.1 Single-objective packaging algorithm

- **Input:**
  - n: Total number of hoppers ($n > 0$).
  - k: Number of hoppers involved in each packaging operation ($2 \leq k < n$).
  - T: Target weight ($T > 0$).
  - $n_1, \dots, n_3$: Number of hoppers in each hopper subgroup ($n_j \geq 0,\ \forall j = 1, \dots, 3;\ \sum_{j=1}^{3} n_j = n$).
  - σ: Standard deviation of the weights supplied to each hopper ($\sigma > 0$).
  - δ: Shift in the mean weights supplied to hoppers in subgroups 1 and 3 compared with subgroup 2 ($\delta > 0$).
  - Q: Total number of packages to be produced ($Q \geq 1$).
- **Step 1. Initialization.**
  - Assign each hopper to a subgroup, such that the number of hoppers in subgroup j is n_j, for all.
  - Calculate the average weights to supply to each hopper subgroup: $\mu_1 = \mu - \delta\sigma$, $\mu_2 = \mu$ and $\mu_3 = \mu + \delta\sigma$.
  - Initialize the contents of each hopper: $x_i = 0,\ \forall i = 1, \dots, n$.
  - Initialize the number of packages produced so far: $q = 0$.
- **Step 2. New packaging operation.** Initialize $z_1^{\min} = +\infty$, $H'_{\min} = \emptyset$.
- **Step 3. Refill all empty hoppers.** For each hopper i in subgroup j for which $x_i = 0$, let $x_i =$ a random value chosen from the distribution $N(\mu_j, \sigma = \gamma\mu_j)$.
- **Step 4. Evaluate all valid subsets to calculate $\boldsymbol{z_1^{\min}}$.** For all k-hopper subsets H' such that obey $|\, T - \sum_{i \in H'} x_i \,| \leq Z_{\alpha/2}\sqrt{k}\sigma$, proceed as follows.
  - Calculate $z_1 = |\ T - \sum_{i \in H'} X_i\ |$. *(i.e., the difference from the target weight)*
  - If $z_1 < z_1^{\min}$, then $z_1^{\min} = z_1$ and $H'_{\min} = H'$.
- **Step 5. Check that the set of valid subsets is not empty.** If $z_1^{\min} = +\infty$ (i.e., there are no valid subsets) then all hoppers must be emptied and refilled. If so, let $x_i = 0$ for each hopper i, then go to Step 2. Otherwise, continue to Step 6.
- **Step 6. Select the $\boldsymbol{k}$-hopper combination that minimizes $\boldsymbol{|\, T - \sum_{i \in H'} x_i \,|}$.** Return $H'_{\min}$ (as the hopper subset for creating the $(q + 1)$-th package). Then, for each hopper i in $H'_{\min}$, let $x_i = 0$ (as it has been emptied to create the package).
- **Step 7. Update the number of packages produced and check whether the process is complete.** Let $q = q + 1$. If $q < Q$ then go to Step 2. Otherwise, END.

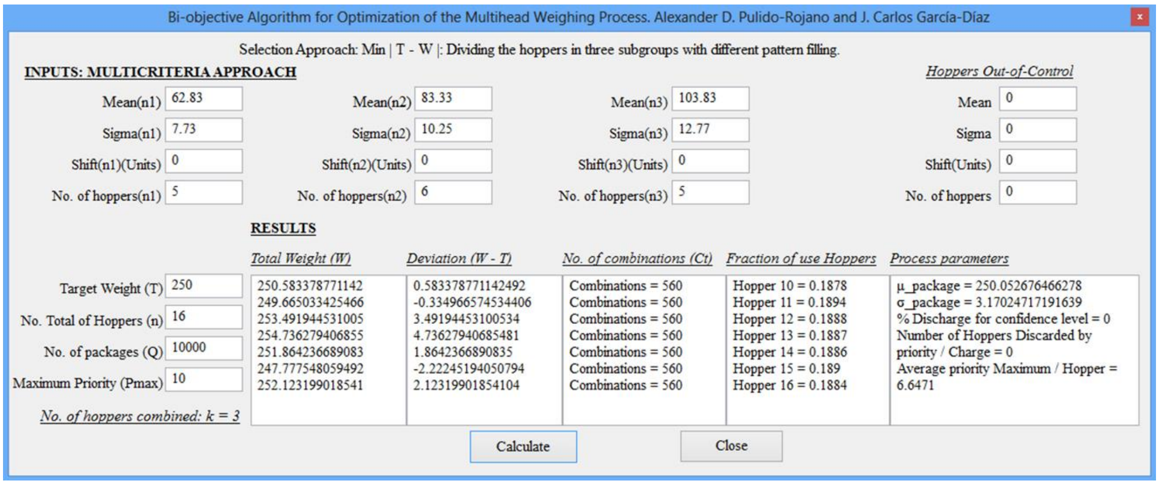
3.3.2 Bi-objective packaging algorithm

- **Input:**
  - n: Total number of hoppers ($n > 0$).
  - k: Number of hoppers involved in each packaging operation ($2 \leq k < n$).
  - T: Target weight ($T > 0$).
  - $n_1, \dots, n_3$: Number of hoppers in each hopper subgroup ($n_j \geq 0$, $\forall j = 1, \dots, 3$; $\sum_{j=1}^{3} n_j = n$).
  - σ: Standard deviation of the weights supplied to each hopper ($\sigma > 0$).
  - δ: Shift in the mean weights supplied to hoppers in subgroups 1 and 3 compared with subgroup 2 ($\delta > 0$).
  - P_{max}: Maximum allowed priority (number of iterations without being chosen) for any hopper ($P_{max} \geq 1$).
  - Q: Total number of packages to be produced ($Q \geq 1$).
- **Step 1. Initialization.**
  - Assign each hopper to a subgroup, so that the number of hoppers in subgroup j is n_j, for all.
  - *Calculate the average weights to supply to each hopper subgroup:* $\mu_1 = \mu - \delta\sigma$, $\mu_2 = \mu$ and $\mu_3 = \mu + \delta\sigma$.
  - *Initialize the contents and priorities for each hopper:* $x_i = 0,\ P_i = 0,\ \forall i = 1, \dots, n$.
  - *Initialize the number of packages produced so far:* $q = 0$.
- **Step 2. New packaging operation.** Initialize $z_1^{\min} = +\infty$, $z_1^{\max} = -\infty$, $z_2^{\min} = +\infty$, $z_2^{\max} = -\infty$, $D_{\min} = +\infty$, $H'_{\min} = \emptyset$.
- **Step 3. Refill all empty hoppers and update priorities.** For each hopper i in subgroup j for which $x_i = 0$, let $x_i =$ a random value chosen from the distribution $N(\mu_j, \sigma = \gamma\mu_j)$. Then, for each hopper i, let $P_i = P_i + 1$.
- **Step 4. Empty any hopper that does not meet the priority constraint.** For each hopper i such that $P_i > P_{max}$, let $x_i = 0,\ P_i = 0$.
- **Step 5. Evaluate of all valid subsets to calculate $\mathbf{z_1^{min}}$, $\mathbf{z_1^{max}}$, $\mathbf{z_2^{min}}$, $\mathbf{z_2^{max}}$.** For each k-hopper subset H' such that does not contain a hopper i with $P_i = 0$ and obeys $|\ T - \sum_{i \in H'} X_i\ | \leq Z_{\alpha/2}\sqrt{k}\sigma$, proceed as follows.
  - Calculate $z_1 = |\ T - \sum_{i \in H'} X_i\ |$. *(i.e., the difference from the target weight)*
  - Calculate $z_2 = \sum_{i \in H'} P_i$. *(i.e., the sum of priorities)*
  - If $z_1 < z_1^{\min}$, then $z_1^{\min} = z_1$.
  - If $z_1 > z_1^{\max}$, then $z_1^{\max} = z_1$.
  - If $z_2 < z_2^{\min}$, then $z_2^{\min} = z_2$.
  - If $z_2 > z_2^{\max}$, then $z_2^{\max} = z_2$.
- **Step 6. Check that the set of valid subsets is not empty.** If $z_1^{\min} = +\infty$ (i.e., there are no valid subsets) then all hoppers must be emptied and refilled. If so, let $x_i = 0, P_i = 0$ for each hopper i, then go to Step 2. Otherwise, continue to Step 7.
- **Step 7. Calculate $\boldsymbol{\theta} = \frac{1}{P_{max} - \max_{i \in H} P_i + 1}$, where $\boldsymbol{H}$ is the set of all hoppers.** This sets the relative importance of the priority objective, and is recalculated before each packaging operation.

- **Step 8. Evaluate all valid subsets again, and select the one that minimizes the performance function D.** For each k-hopper subset H' that does not contain a hopper i with $P_i = 0$ and obeys $| T - \sum_{i \in H'} X_i \ | \leq Z_{\alpha/2}\sqrt{k}\sigma$, proceed as follows.
    - Retrieve the z_1 and $z_2$values that were calculated for H' at Step 5.
    - Calculate $D = \sqrt{(1-\theta)\left(\frac{z_1 - z_1^{\min}}{z_1^{\max} - z_1^{\min}}\right)^2 + \theta\left(\frac{z_2 - z_2^{\max}}{z_2^{\max} - z_2^{\min}}\right)^2}$.
    - If $D < D_{\min}$, then $D_{\min} = D$, $H'_{\min} = H'$.
- **Step 9. Select the k-hopper subset that minimizes D.** Return $H'_{\min}$ as the hopper subset for creating the $(q+1)$-th package. For each hopper i in $H'_{\min}$, let $x_i = 0, P_i = 0$ (as it has been emptied to create the package).
- **Step 10. Update the number of packages produced and check whether the process is complete.** Let $q = q + 1$. If $q < Q$ then go to Step 2. Otherwise, END.

In these enumerative (exhaustive) algorithms, every feasible solution (i.e., valid k-hopper subset) is evaluated at each iteration. In particular, the number of subsets to be evaluated for each packaging operation is at most $\binom{n}{k} = n!/(k!\,(n-k)!)$ (less for the bi-objective algorithm when hoppers are discarded at Step 4 due to the priority constraint). Although it is a simple strategy, it means our algorithms conduct exact (not heuristic) searches (Michalewicz and Fogel, 2004).

Both algorithms consider the situation where all hoppers have to be emptied to avoid producing packages that would not meet the quality requirements in terms of weight. However, the emptied products could, for example, be taken and reused. In addition, we will calculate how often this happens, which we call the *confidence level* (DCL), as a performance measure. Figure 2 shows the application's user interface for the bi-objective case and k = 3.



Bi-objective Algorithm for Optimization of the Multihead Weighing Process. Alexander D. Pulido-Rojano and J. Carlos García-Díaz

Selection Approach: Min | T - W |: Dividing the hoppers in three subgroups with different pattern filling.

INPUTS: MULTICRITERIA APPROACH

Mean(n1)	62.83	Mean(n2)	83.33	Mean(n3)	103.83	Mean	0
Sigma(n1)	7.73	Sigma(n2)	10.25	Sigma(n3)	12.77	Sigma	0
Shift(n1)(Units)	0	Shift(n2)(Units)	0	Shift(n3)(Units)	0	Shift(Units)	0
No. of hoppers(n1)	5	No. of hoppers(n2)	6	No. of hoppers(n3)	5	No. of hoppers	0

Hoppers Out-of-Control

Target Weight (T) 250
No. Total of Hoppers (n) 16
No. of packages (Q) 10000
Maximum Priority (Pmax) 10
No. of hoppers combined: k = 3

RESULTS

Total Weight (W)	Deviation (W - T)	No. of combinations (Ct)	Fraction of use Hoppers
250.583378771142	0.583378771142492	Combinations = 560	Hopper 10 = 0.1878
249.665033425466	-0.334966574534406	Combinations = 560	Hopper 11 = 0.1894
253.491944531005	3.49194453100534	Combinations = 560	Hopper 12 = 0.1888
254.736279406855	4.73627940685481	Combinations = 560	Hopper 13 = 0.1887
251.864236689083	1.8642366890835	Combinations = 560	Hopper 14 = 0.1886
247.777548059492	-2.22245194050794	Combinations = 560	Hopper 15 = 0.189
252.123199018541	2.12319901854104	Combinations = 560	Hopper 16 = 0.1884

Process parameters

µ_package = 250.052676466278
σ_package = 3.17024717191639
% Discharge for confidence level = 0
Number of Hoppers Discarded by priority / Charge = 0
Average priority Maximum / Hopper = 6.6471

Figure 2. Our application's user interface, for the bi-objective case and k = 3.
Source: Authors.

3.4. Mathematical optimisation

Solutions to the packaging problem can be described in terms of binary vectors. For three groups, we have $[y_1^1, \ldots, y_{n1}^1]$, $[y_1^2, \ldots, y_{n2}^2]$ y $[y_1^3, \ldots, y_{n3}^3]$, where the value of each component indicates whether the corresponding weight was selected (1) or not (0). In terms of these vectors, the solutions to the bi-objective problem can be described as follows:

$$y_i^1 = \begin{cases} 1 \; if\ weight\ \ i \in n_1\ has\ been\ chosen \\ \quad 0 \qquad otherwise \end{cases}$$

$$y_l^2 = \begin{cases} 1 \; if\ weight\ l \in n_2\ has\ been\ chosen \\ \quad 0 \qquad otherwise \end{cases}$$

$$y_r^3 = \begin{cases} 1\, if\ weight\ \ r \in n_3\ has\ been\ chosen \\ \quad 0 \qquad otherwise \end{cases}$$

Objective 1 → $Minimize\ \ z_1$

Objective 2 → $Maximize\ \ z_2 = \sum_{i=1}^{n_1} P_i^1 y_i^1 + \sum_{l=1}^{n_2} P_l^2 y_l^2 + \sum_{r=1}^{n_3} P_r^3 y_r^3$

Subject to

$$z_1 \ \geq 0 \tag{3}$$

$$z_1 \ \geq \ T - \sum_{i=1}^{n_1} x_i^1 y_i^1 + \sum_{l=1}^{n_2} x_l^2 y_l^2 + \sum_{r=1}^{n_3} x_r^3 y_r^3 \tag{4}$$

$$z_1 \ \geq \sum_{i=1}^{n_1} x_i^1 y_i^1 + \sum_{l=1}^{n_2} x_l^2 y_l^2 + \sum_{r=1}^{n_3} x_r^3 y_r^3 - T \tag{5}$$

$$T - Z_{\alpha/2}\sqrt{k}\sigma \leq \sum_{i=1}^{n_1} x_i^1 y_i^1 + \sum_{l=1}^{n_2} x_l^2 y_l^2 + \sum_{r=1}^{n_3} x_r^3 y_r^3 \tag{6}$$

$$T + Z_{\alpha/2}\sqrt{k}\sigma \geq \sum_{i=1}^{n_1} x_i^1 y_i^1 + \sum_{l=1}^{n_2} x_l^2 y_l^2 + \sum_{r=1}^{n_3} x_r^3 y_r^3 \tag{7}$$

$$\forall\ P_{i \in H'}^1 \leq P_{max} \tag{8}$$

$$\forall\ P_{l \in\, H'}^2 \leq P_{max} \tag{9}$$

$$\forall\ P_{r \in H'}^3 \ \leq P_{max} \tag{10}$$

$$y_i^1 \in \{0, 1\}, \quad i = 1,2, \dots, n_1. \tag{11}$$

$$y_l^2 \in \{0, 1\}, \quad l = 1,2, \dots, n_2. \tag{12}$$

$$y_r^3 \in \{0, 1\}, \quad r = 1,2, \dots, n_3. \tag{13}$$

Eqs. (3) – (5) ensure that z_1 is non-negative. The *confidence level constraint* is represented by Eqs. (6) and (7). Eqs. (8) – (10) ensure that the k-hopper subset selected to form H' does not exceed the maximum allowed priority. The binary constraints on the variables y_i^1, y_l^2 and y_r^3 are represented by Eqs. (11) – (13). This model can be adapted for the single-objective approach by only considering the objective z_1 and omitting the constraints expressed in Eqs. (8) – (10).

4. Results and analysis

Here, we use a series of numerical experiments to demonstrate the effectiveness of our algorithms for reducing package weight variability, in terms of the most important performance parameters for the packaging process: The average weight μ_{package} and standard deviation σ_{package} of the packages produced, the number of hoppers emptied for package weight reasons (i.e., the DCL), the number of hoppers emptied for priority reasons at each iteration (which we call the HDP), and the average maximum priority for each hopper (which we call the AMP).

The parameters for the first experiment were as followings: the total number of hoppers n = 16; the number of hoppers in each subset k: ranged from 2 – 8; the target weight T = 250 g; the sizes of the hopper subgroups were n_1 = 5, n_2 = 6, and n_3 = 5; the proportionality coefficients γ were 0.123 and 0.331; the shift value δ = 2.0; and the maximum allowed priorities P_{max} were 10, 50, and 100.

Table 1 shows the results when these packaging process parameters were used on the evaluated approaches (single-objective and bi-objective). To show the effectiveness of our algorithms more clearly, results for a single-objective optimisation approach that does not divide the hoppers into subgroups are presented for comparison. It should also be emphasized that the results of our experiments are not compared with García-Díaz et al. (2017), because, as we have already highlighted, these authors assumed the σ values to each hopper and used different input values, leading to very different outcomes.

The results show that, for the bi-objective approach, the σ_{package} values decreased and the APM values increased as the P_{max} value increased. The σ_{package} values were only greater than the expected value $\sqrt{k}\sigma$ for k = 2, regardless of the product type.

For γ = 0.123, the AMP values decreased as k is increased, and the HDP was always zero, except for k = 2 and P_{max} = 10. The optimal value of k was 7.

For γ = 0.331, σ_{package} was minimized for k = 7 (for P_{max} = 50 or 100) and k = 8 (for P_{max} = 10). There were also cases where increases in k and P_{max} produced increases in the σ_{package} , HDP, and AMP values. Taken together, these results show that increases in γ can lead to hopper weights that are difficult to combine, affecting the variability of the process.

When k = 2, the base line single-objective approach (where the hoppers were not divided into subgroups) produced the best σ_{package} results. In all other cases, however, our alternative approaches, with the right k, were able to improve process variability.

In general, the best results were obtained by the single-objective approach with three hopper subgroups and k = 6 (for γ = 0.123) or k = 5 (for γ = 0.331). However, the highest AMP values were obtained by this approach with k = 2. In addition, using high k values (up to k = 8) did not guarantee the lowest σ_{package} values. Products were never emptied from hoppers due to the *confidence level constraint* for any of the approaches tested.

5. Conclusions

Optimisation allows us to discover the best alternatives for problems that can be modeled mathematically and is fundamental to improving the quality of industrial processes. In this paper, we have presented both a single-objective approach and a bi-objective approach to optimising the multihead weighing process. These approaches were expressed in terms of both mathematical models and algorithms.

Table 1. Results for packaging parameters μ_{package}, σ_{package} , DCL(%), HDP and APM for P_{max}: {10,50,100}, k:{2,3,4,5,6,7,8}, γ: {0.123,0.331}, δ:{2.0}, and n:{16}.

			Bi-objective approach (3 hopper subgroups)															**Single-objective approach (3 hopper subgroups)**				**Single-objective approach (all hoppers in one group)**			
			P_{max} = 10					**P_{max} = 50**					**P_{max}= 100**												
γ	k	$\sqrt{k}\sigma$	μ_{package}	σ_{package}	DCL	HDP	APM	μ_{package}	σ_{package}	DCL	HDP	APM	μ_{package}	σ_{package}	DCL	HDP	APM	μ_{package}	σ_{package}	DCL	APM	μ_{package}	σ_{package}	DCL	APM
0.123	2	21.74	250.22	27.83	0.00	0.0044	9.02	252.71	23.12	0.00	0.00	19.67	253.33	22.21	0.00	0.00	28.94	249.27	6.72	0.00	3955.77	249.72	4.08	0.00	289.27
	3	17.75	250.05	3.17	0.00	0.00	6.64	249.99	1.09	0.00	0.00	7.40	250.00	0.834	0.00	0.00	7.75	250.00	0.198	0.00	33.07	249.74	3.07	0.00	80.94
	4	15.38	249.99	1.48	0.00	0.00	5.11	250.00	0.642	0.00	0.00	5.35	250.01	0.490	0.00	0.00	5.45	249.99	0.035	0.00	14.32	249.87	2.92	0.00	41.60
	5	13.75	249.99	0.967	0.00	0.00	4.14	250.00	0.448	0.00	0.00	4.27	250.00	0.343	0.00	0.00	4.33	250.00	0.011	0.00	10.47	249.86	3.04	0.00	30.87
	6	12.55	249.99	0.684	0.00	0.00	3.45	249.99	0.337	0.00	0.00	3.63	249.99	0.264	0.00	0.00	3,69	250.00	0.0051	0.00	8.16	249.91	3.01	0.00	21.33
	7	11.62	249.99	0.602	0.00	0.00	3.06	249.99	0.262	0.00	0.00	3.11	249.99	0.201	0.00	0.00	3.13	249.99	0.011	0.00	6.74	249.83	3.04	0.00	16.60
	8	10.87	249.99	0.615	0.00	0.00	2.94	250.00	0.314	0.00	0.00	2.98	250.00	0.242	0.00	0.00	2.99	249.99	0.012	0.00	5.58	249.92	3.09	0.00	13.76
0.331	2	58.51	253.61	82.27	0.00	0.0082	9.05	273.15	80.70	0.00	0.0026	22.22	271.29	77.09	0.00	0.0026	35.71	243.88	24.73	0.00	4613.31	249.24	10.98	0.00	289.27
	3	47.78	251.54	15.72	0.00	0.0039	6.83	250.74	8.61	0.00	0.0039	16.81	250.59	8.17	0.00	0.0039	39.15	249.76	6.01	0.00	2994.73	249.33	8.31	0.00	87.54
	4	41.38	250.11	6.99	0.00	0.0055	5.24	250.22	6.23	0.00	0.0055	10.78	250.20	5.90	0.00	0.0055	26.65	249.99	0.171	0.00	157.14	249.66	7.85	0.00	41.60
	5	37.01	250.00	6.18	0.00	0.0072	4.32	250.07	5.40	0.00	0.0072	11.41	250.19	5.74	0.00	0.0071	30.79	250.00	0.036	0.00	11.02	249.63	8.17	0.00	30.87
	6	33.78	250.27	6.38	0.00	0.0083	3.71	250.28	5.99	0.00	0.0082	11.93	250.24	6.09	0.00	0.0082	33.81	249.99	0.434	0.00	8.24	249.73	8.18	0.00	20.82
	7	31.28	250.07	5.27	0.00	0.0095	3.34	250.17	5.40	0.00	0.0095	12.57	250.22	5.48	0.00	0.0094	36.65	249.99	0.088	0.00	6.72	249.59	8.17	0.00	18.23
	8	29.26	250.18	5.24	0.00	0.0108	3.24	250.28	5.86	0.00	0.0107	13.49	250.12	6.35	0.00	0.0106	39.37	249,99	0.322	0.00	5.67	249.78	8.32	0.00	13.76

The single-objective approach aims to minimize the absolute difference between each real package weight and the target weight, while the bi-objective approach also aims to maximize the total priority of the chosen hopper subset. Both approaches aim to improve the quality of the packages produced by adjusting the way the weighing hoppers are filled and dividing them in subgroups.

Bi-objective algorithm combines information about these two objectives, dynamically adjusting the relative weight or importance of each objective for each iteration (packaging operation).

We have also investigated the effectiveness of the solutions produced by our approaches, comparing them with a more traditional approach. Based on these results, we have concluded that dividing the hoppers into subgroups is effective even when there is a limit to how long each load can be allowed to remain in its hopper. The average highest observed priority (AMP) for the bi-objective algorithm was significantly lower than for the single-objective approach. However, we have found that products with a high coefficient of proportionality affect process variability and increase the average time loads spend in the hoppers. We also found that using large hopper subsets to create each package does not guarantee reduced variability.

For the setup of the process and guarantee the least variability, we recommend in the single-objective approach dividing the hoppers in three subgroups with n_1 = 5, n_2 = 6, n_3 = 5, δ = 2.0, P_{max} = 100 and k = 6 (for γ = 0.123) or k = 5 (for γ 0.331). For the bi-objective approach, we recommend dividing the hoppers in three subgroups with n_1 = 5, n_2 = 6, n_3 = 5, δ = 2.0, P_{max} = 100 and k = 7 (for γ = 0.123) or P_{max} = 10 and k = 8 (for γ = 0.331).

In future research, we propose to study further the reason for the increase in the σ_{package} , HDP, and AMP values when the γ, k and P_{max} values increase. In addition, we propose to implement different multi-objective optimisation approach for this problem, considering different packaging algorithms and objectives of economic character (product packaging costs, cost of rejection and rework of the "non- conforming" package). Likewise, we intend to study the relationships among all the factors that may influence the packaging process.

References

Barreiro, J.J., González, C. and Salicrú, M. (1998) 'Optimization of multiweighing packing proceeding', *Top*, Vol. 6, No. 1, pp.37–44.

Beretta, A. and Semeraro, Q. (2012), 'On a RSM approach to the multihead weigher configuration' *in the 11th biennial Conference on Engineering Systems Design and Analysis (ASME 2012)*, Nantes, France, pp. 225-233.

Beretta, A., Semeraro, Q. and del Castillo E. (2016) 'On the Multihead Weigher Machine Setup Problem'. *Packaging Technology and Science*, Vol. 29, No. 3, pp. 175–188.

Bierlaire, M. (2015) *Optimization: principles and algorithms*, 1st edition, EPFL Press, Lausana.

Blum, C., Blesa-Aguilera, M.J., Roli, A. and Sampels, M. (2008*) Hybrid Metaheuristics: An Emerging Approach to Optimization*, Springer-Verlag, Berlin.

Cook, W., Cunningham, W., Pulleyblank, W. and Schrijver, A. (1998) *Combinatorial Optimization*, John Wiley & Sons, New York.

del Castillo, E., Beretta, A. and Semeraro, Q. (2017) 'Optimal setup of a multihead weighing machine'. *European Journal of Operational Research*, Vol. 259, No. 1, pp. 384 - 393.

Duarte, A., Pantrigo, J.J. and Gallego Carrillo, M. (2007) *Metahuerísticas*, Dykinson, Madrid.

Ehrgott, M. (2005) *Multicriteria Optimization*, 2nd ed., Springer, Berlin

Erdogdu, F. (2009) *Optimization in Food Engineering*, Taylor and Francis, London.

García-Díaz, J.C. and Pulido-Rojano, A. (2017) 'Monitoring and control of the multihead weighing process through a modified control chart', *DYNA*. Vol. 84. No. 200, pp. 135-142.

García-Díaz, J.C., Pulido-Rojano A. and Giner-Bosch, V. (2017) 'Bi-objective optimization of a multihead weighing process'. *European Journal Industrial Engineering*, Vol. 11, No. 3, pp. 403–423.

Garey, M.R. and Johnson, D.S. (1979) *Computers and Intractability: A Guide to the Theory of NP-Completeness*, WH Freeman and Company, New York.

Imahori, S., Karuno, Y., Nagamochi, H. and Wang, X. (2011) 'Kansei engineering humans and computers: efficient dynamic programming algorithms for combinatorial food packing problems', *International Journal of Biometrics*, Vol. 3, No. 3, pp. 228–245.

Imahori, S., Karuno, Y., Nishizaki, R. and Yoshimoto, Y. (2012) 'Duplex and quasi-duplex operations in automated food packing systems' *in 2012 IEEE/SICE International Symposium on System Integration (SII),* Fukuoka, Japan, pp.810–815.

Karuno, Y., Nagamochi, H. and Wang, X. (2007) 'Bi-criteria food packing by dynamic programming', Journal of the Operations Research Society of Japan, Vol. 50, No. 4, pp.376–389.

Karuno, Y., Nagamochi, H. and Wang, X. (2010) 'Optimization problems and algorithms in double-layered food packing systems', Journal of Advanced Mechanical Design, System, and Manufacturing, Vol. 4, No. 3, pp. 605–615.

Karuno, Y., Takahashi, K. and Yamada, A. (2013) 'Dynamic programming algorithms with data rounding for combinatorial food packing problems', *Journal of Advanced Mechanical Design, System, and Manufacturing*, Vol. 7, No. 2, pp.233–243.

Karuno, Y. and Tateishi, K. (2014) 'Improved heuristics with data rounding for combinatorial food packing problems' *in IEEE 2014 7th International Conference on Service-Oriented Computing and Applications*, Matsue, Japan, pp.81–88.

Karuno Y. and Saito, R. (2017) 'Heuristic algorithms with rounded weights for a combinatorial food packing problem', Journal of Advanced Mechanical Design, System, and Manufacturing, Vol. 11, No. 1, DOI: https://doi.org/10.1299/jamdsm.2017jamdsm0003.

Keraita, J.N. and Kim, K-H. (2006) 'A study on the optimum scheme for determination of operation time of line feeders in automatic combination weighers', *Journal of Mechanical Science and Technology*, Vol. 20, No. 10, pp.1567–1575.

Keraita, J.N. and Kim, K-H. (2007) 'A weighing algorithm for multihead weighers', *International Journal of Precision Engineering and Manufacturing*, Vol. 8, No. 1, pp.21–26.

Marler, T. (2009) Multi-Objective Optimization: Concepts and Methods for Engineering, VDM Verlag, Saarbruckend, Germany.

Michalewicz, Z. and Fogel, D.B. (2004) *How to Solve It: Modern Heuristics*, 2nd ed., Springer Science & Business Media, Berlin.

Montgomery, D.C. (2009). *Introduction to Statistical Quality Control*, 6th ed., John Wiley and Sons, New York, NY.

Nemhauser, G. and Wolsey, L. (1988) *Integer and Combinatorial Optimization*, John Wiley & Sons, New York.

Pulido-Rojano, A., García-Díaz, J.C. and Giner-Bosch, V. (2015) 'A multiobjective approach for optimization of the multihead weighing process', in Framinan J.M., Perez P. and Artiba, A. (eds), The Road Ahead: Understanding Challenges and Grasping Opportunities in Industrial and Systems Engineering, IEEE: Spain, p. 67.

Pulido-Rojano, A. and García-Díaz, J.C. (2016), 'Analysis of the Filling Setting in the Multihead Weighing Process' in Proceedings of the International Joint Conference - CIOICIEOM- IIE-AIM (IJC 2016), San Sebastián, Spain, pp. 521 - 528.

Salicrú, M., González, C. and Barreiro, J.J. (1996) 'Variability reduction with multiweighing proceedings', *Top*, Vol. 4, No. 2, pp.319–329.

Sasadhar, B. and Indrajit, M. (2018) 'Advances in solution methods for optimisation of multiple quality characteristics in manufacturing processes', International Journal of Productivity and Quality Management, Vol.24, No.4, pp. 475 – 494. DOI: 10.1504/IJPQM.2018.093448

Selvam, G., Arockia S., Surya V. and Rohit, T. (2018) 'Quality and productivity improvement through spot welding process optimisation in automobile body shop', International Journal of Productivity and Quality Management, Vol.23, No.1, pp.110 – 127. DOI: 10.1504/IJPQM.2018.088611

Sukrut, N. and Mohammed, I. (2017) 'Quality enhancement through first pass yield using statistical process control', International Journal of Productivity and Quality Management, Vol. 20, No. 2, pp. 238 – 253. DOI: 10.1504/IJPQM.2017.081482

Tejaskumar, S.P. and Darshak, A.D. (2016) 'A case study: to reduce process variability of valve seat depth in cylinder head using Six Sigma methodology', International Journal of Productivity and Quality Management, Vol.17, No. 4, pp. 474 - 506. DOI: 10.1504/IJPQM.2016.075250